

## 1. 목적

유닉스 시스템에 대하여 미리 접해본 후 실험에 임할 수 있도록 한다. 아울러 부록에 나와 있는 명령어에 대하여 익숙해지도록 사용해본다.

## 2. 예비 학습

UNIX 시스템에 접속해본 뒤 자신의 홈 디렉토리를 확인해본다.

```
cse20180305@cspro1:~$ pwd
/sogang/under/cse20180305
```

셸 프로그래밍 실험에서 사용할 데이터 파일인 전화번호부를 만들어본다. 단 데이터 파일의 형식은 실험에서 나온 예제에 따르도록 한다. 5명 이상이 들어가 있는 데이터를 만들되 vi 에디터를 이용하여 작성한다. 단 파일명은 data로 한다.

### data

```
홍길동|서울시 마포구 신수동 서강대학교 AS관 301호|02-705-2665
박문수|서울시 서대문구 신촌동 연세대학교 연구관 102호|02-708-4678
Andrew|경기도 의정부시 호원동 23-12번지|031-827-7842
한승현|경기도 고양시 일산동구 장항2동 호수마을 407호 1602호|031-904-5643
신봉선|경기도 고양시 일산동구 마두동 206동|031-813-3803
```

위의 예제를 편집하는데 사용한 vi 명령어들을 나열하고, 해당 명령 수행하는 결과를 적어보도록 한다.

```
vi (파일 이름) : 텍스트 편집기 vi로 파일이 실행된다.
i (vi 명령어) : vi를 입력 모드로 전환한다.
(ESC) :wq : 작성한 파일을 저장한 후 vi를 종료한다.
```

위에서 작성한 데이터 파일을 \$home/.data 파일로 복사한다. 복사하기 위하여 사용한 명령들을 적어보도록 한다.

```
파일 복사를 위해 명령어 cp를 사용한다.
cse20180305@cspro1:~$ cp data $HOME/.data
```

여기서 .파일은 숨김파일이다. 복사 후 홈 디렉토리에서 ls를 하면 변화가 없지만,  
ls -al을 하면 파일명이 .으로 시작하는 숨김파일을 확인할 수 있다.

\$HOME/.data 파일을 그룹 및 다른 사용자가 아무 권한도 갖지 않도록 권한 변경을 해본다. 사용한 명령을 적어보도록 한다.

chmod는 change mode의 축약어로 대상 파일과 디렉토리의 권한 변경에 쓰인다.  
**cse20180305@cspro1:~\$ chmod 777 \$HOME/.data**

디렉토리에 대한 읽기, 쓰기, 실행 권한을 설정해보고 각각이 갖는 의미를 살펴본다.

+r (read/읽기): 디렉토리 안의 파일을 읽을 수 있다. (ls 명령어 등 사용 가능하다.)  
+w (write/쓰기): 디렉토리 안의 파일이나 디렉토리를 작성, 수정, 삭제할 수 있다.  
+x (execute/실행): 디렉토리에 접근할 수 있다. (cd 명령어 등 사용 가능하다.)  
이 권한 없이는 읽기와 쓰기 권한도 전부 불가능하다.

### 3. 보충 학습

Regular Expression에 대하여 정리해보도록 한다.

정규표현식(Regular Expression)은 특정한 규칙을 가진 문자열의 집합을 표현하기 위해 쓰이는 형식 언어이다. 줄여서 Regex, Regexp 등으로 불린다.  
정규표현식에는 다양한 형태가 있지만 크게 UNIX 계열에서 사용하는 POSIX 표준과 Perl에서 유래한 PCRE(Perl Compatible Regular Expressions)가 있다.

정규식의 기본 문법의 크게 3가지로 나눌 수 있다.

1. 패턴 그대로를 매칭 : 단어 그대로를 패턴으로 사용하여 매치하는 영역을 찾는다.
2. 메타 문자 및 수량 한정자 : 보다 폭넓은 패턴에 매치할 때 사용한다.
3. 그룹 및 look around 기능 : 패턴의 일부를 그룹화하거나, 특정 패턴의 앞 뒤로 다른 패턴이 오는 조건을 더하는 경우다.

정규식 메타 문자를 이용하면 특정한 규칙을 가진 여러 단어를 하나의 패턴으로 함축할 수 있다.

1. ^ 문자열의 시작. [...] 내에서 쓰이면 '일치하지 않는'의 의미가 된다.
2. \$ 문자열의 끝
3. \wb 단어의 경계. 공백, 탭, 콤마, 대시 등이 올 수 있다.
4. \WB \wb 가 아닌 것. 정규식 메타문자에서는 대소문자가 바뀌면 반대의 의미를 지니는 것들이 있다.
5. \WS 공백 문자가 아닌 것
6. \wd 숫자. [0-9]와 같다.
7. \WD 숫자가 아닌 것. [^0-9]와 같다.
8. \ww 단어를 만들 수 있는 글자. 알파벳 대소문자, 숫자, 언더 스코어가 포함된다.
9. \WW not \ww . 위의 것이 아닌 글자들이 해당된다.
10. \wn 개행 문자. 캐리지리턴은 \wr 에 매치한다.
11. \w 이스케이프용 문자. 정규식 상의 특별한 의미가 있는 기호들을 문자 그대로 사용할 때 쓴다.
12. . 임의의 문자 1 개에 대응한다.

등등